

⑫ 公開特許公報 (A) 平3-176148

⑮ Int. Cl.⁵
 B 41 B 23/00
 27/00
 G 06 F 15/20

識別記号
 5 3 6

序内整理番号
 8530-2H
 8530-2H
 7165-5B

⑯ 公開 平成3年(1991)7月31日

審査請求 未請求 請求項の数 6 (全18頁)

⑭ 発明の名称 文書レイアウト編集装置

⑮ 特願 平1-315213

⑮ 出願 平1(1989)12月6日

⑯ 発明者	是枝 浩行	神奈川県横浜市戸塚区吉田町292番地 株式会社日立製作所マイクロエレクトロニクス機器開発研究所内
⑯ 発明者	桑原 梅司	神奈川県横浜市戸塚区吉田町292番地 株式会社日立製作所マイクロエレクトロニクス機器開発研究所内
⑯ 発明者	野中 尚道	神奈川県横浜市戸塚区吉田町292番地 株式会社日立製作所マイクロエレクトロニクス機器開発研究所内
⑯ 発明者	中根 啓一	神奈川県横浜市戸塚区吉田町292番地 株式会社日立製作所マイクロエレクトロニクス機器開発研究所内
⑮ 出願人	株式会社日立製作所	東京都千代田区神田駿河台4丁目6番地
⑮ 代理人	弁理士 小川 勝男	外1名

明細書

1. 発明の名称

文書レイアウト編集装置

2. 特許請求の範囲

1. 少なくとも演算処理装置と、主メモリと、表示装置と、入力装置とを有する計算機装置において、複数の文書構成要素データからなる多段組み文書のレイアウトを行う文書レイアウト編集装置であって、

指定された文書構成要素データの量および書式に従って、該文書構成要素データを割付けるのに必要な大きさの枠を仮表示する枠表示手段と、上記枠表示手段により仮表示された枠に対する領域と、すでに設定済みの既存領域とが重ならないよう割付けを行う割付け手段とを設けたことを特徴とする文書レイアウト編集装置。

2. 上記枠の大きさを変えずに、該枠の形状を変更する枠形状変更手段と、段単位で、該枠を移動する枠移動手段とを設けたことを特徴とする請求項1記載の文書レイアウト編集装置。

3. 上記割付け手段は、上記枠に対応する領域に重なる既存領域を、領域同士が重ならないように、形状を変更することを特徴とする請求項1記載の文書レイアウト編集装置。

4. 上記割付け手段は、上記既存領域に重なる上記枠に対応する領域を、領域同士が重ならないように、形状を変更することを特徴とする請求項1記載の文書レイアウト編集装置。

5. 少なくとも演算処理装置と、主メモリと、表示装置と、入力装置とを有する計算機装置において、複数の文書構成要素データからなる多段組み文書のレイアウトを行う文書レイアウト編集装置であって、

関連して割付けるべき文書構成要素データを、ひとまとまりの単位として管理し、文書を上記単位の集りとして管理する管理手段と、指定された上記単位に含まれる上記文書構成要素データについて、順次、各文書構成要素データの量および書式に従って、該文書構成要素データを割付けるのに必要な大きさの枠を仮表示する枠

表示手段と、上記枠表示手段により仮表示された枠に対応する領域と、すでに設定済みの既存領域とが重ならないよう割付けを行う割付け手段とを設けたことを特徴とする文書レイアウト編集装置。

6. 少なくとも演算処理装置と、主メモリと、表示装置と、入力装置とを有する計算機装置において、複数の文書構成要素データからなる多段組み文書のレイアウトを行う文書レイアウト編集装置であって、

関連して割付けるべき文書構成要素データを、ひとまとめりの単位として管理し、文書を上記単位の集りとして管理する管理手段と、上記単位に割付け順序の優先度情報を追加する追加手段と、上記追加手段により追加された優先度情報の順に、上記単位に含まれる文書構成要素データについて、順次、各文書構成要素データの量および書式に従って、該文書構成要素データを割付けるのに必要な大きさの枠を仮表示する枠表示手段と、上記枠表示手段により仮表示さ

し込む場合には、複数の領域22を指定し、その中で、文章を流し込む順番を指定することで、指定された領域22順に1つの文章が流し込まれ、複数段に渡る文章の配置を行うことができる。

第2図の例では、領域22-0には見出し文字のデータを割付けている。見出し文字データは、通常の文章データとは異なり、段にまたがり、領域22内の文字列を配置できる。

また、領域22-1, 22-2, 22-4の順に、1文章を流し込むよう指定し、領域22-5, 22-7の順に、別の文章を、領域22-6, 22-9の順に、さらに別の文章を流し込むよう指定している。領域22-3には図形、領域22-8にはグラフを割当てており、縦書きの新聞のような多段組み文書の1頁を構成することができる。

[発明が解決しようとする課題]

上記従来技術においては、文章を流し込む以前に、文章、図、見出し等の各構成要素データに対する領域の割り振りを考え、領域の配置を決定してから、文章以外の構成要素データを割付け、最

れた枠に対応する領域と、すでに設定済みの既存領域とが重ならないよう割付けを行う割付け手段とを設けたことを特徴とする文書レイアウト編集装置。

3. 発明の詳細な説明

[産業上の利用分野]

本発明は、多段組みされた文書のレイアウトを行う文書レイアウト編集装置に関する。

[従来の技術]

従来、文書のレイアウトを行う組版装置としては、特開昭61-27256号公報に記載されている組版システムを適用したものがあった。

この組版システムにおいては、新聞のように、多段組みで複数の記事等が流し込まれた文書を作るためには、第2図に示すように、画面に表示された用紙20イメージ上で、互いに重ならないように、領域22の枠（図中の実線で囲まれた範囲）を配置して設定し、その後で、文書の構成要素データである文章や写真等を呼出し、各構成要素データを割付ける領域22を指定する。特に、文章を流

し込む場合には、文章を流し込む必要があった。

このような組版システムにおいては、文章に応じた良い配置を決めるためには、ユーザの熟練を要し、文章が指定した領域に入り切らなかったり、領域が大きく余ったりした場合には、もう一度領域の大きさや配置を決め直した後で、再び文章を流し込み直してみる必要があり、作業の煩りが多く、レイアウトに手間がかかるという問題があった。

本発明の目的は、組版ルールを知らない初心者に対しても、容易な操作で、ある程度の美しい段組み文書を作成できる文書レイアウト編集装置を提供することにある。

また、本発明の他の目的は、多段組み文書の各構成要素データを、内容的につながった単位でまとめて、順次レイアウト可能とし、新聞のような文書において、該内容的につながった単位、つまり、記事単位で、レイアウトを決定していくことを容易にすることを目的とする。

[課題を解決するための手段]

上記目的を達成するために、本発明は、少なくとも演算処理装置と、主メモリと、表示装置と、入力装置とを有する計算機装置において、複数の文書構成要素データからなる多段組み文書のレイアウトを行う文書レイアウト編集装置であって、ユーザが指定した文書構成要素データの量および書式に従って、該文書構成要素データを割付けるのに必要な大きさの枠を仮表示する枠表示手段と、ユーザの割付け実行指示に対し、上記枠表示手段により仮表示された枠に対応する領域と、すでに設定済みの既存領域とが重ならないよう割付けを行う割付け手段とを設けている。

また、ユーザが望む位置、段数で、文書構成要素データが割付けられるようにするために、上記に加えて、上記枠表示手段により仮表示された枠に文書構成データが無駄なく収まるよう、該枠に入るデータ量を一定にしたまま、該枠の形状を変更する枠形状変更手段と、段単位で、該枠の移動を行う枠移動手段とを設けることができる。

上記割付け手段は、ユーザが細かな領域配置を

と、上記枠表示手段により仮表示された枠に対応する領域と、すでに設定済みの既存領域とが重ならないよう割付けを行う割付け手段とを設けている。

上記管理手段は、例えば、見出し、要約、本文、写真・図等の文書構成要素データを、1つの記事という単位の中のデータとして管理する。上記枠表示手段および上記割付け手段は、ユーザが記事を指定することにより、その記事の各文書構成要素データについて、連続して処理を行う。

また、本発明は、重要な記事の順に割付けが行なえるようにするために、関連して割付けるべき文書構成要素データを、ひとまとまりの単位として管理し、文書を上記単位の集りとして管理する管理手段と、上記単位に割付け順序の優先度情報を追加する追加手段と、上記追加手段により追加された優先度情報の順に、上記単位に含まれる文書構成要素データについて、順次、各文書構成要素データの量および書式に従って、該文書構成要素データを割付けるのに必要な大きさの枠を仮表

示する枠表示手段と、上記枠表示手段により仮表示された枠に対応する領域と、すでに設定済みの既存領域とが重ならないよう割付けを行う割付け手段とを設けている。

また、上記他の目的を達成するために、本発明は、少なくとも演算処理装置と、主メモリと、表示装置と、入力装置とを有する計算機装置において、複数の文書構成要素データからなる多段組み文書のレイアウトを行う文書レイアウト編集装置であって、関連して割付けるべき文書構成要素データを、ひとまとまりの単位として管理し、文書を上記単位の集りとして管理する管理手段と、指定された上記単位に含まれる文書構成要素データについて、順次、各文書構成要素データの量および書式に従って、該文書構成要素データを割付けるのに必要な大きさの枠を仮表示する枠表示手段

示する枠表示手段と、上記枠表示手段により仮表示された枠に対応する領域と、すでに設定済みの既存領域とが重ならないよう割付けを行う割付け手段とを設けている。

上記管理手段は、例えば、見出し、要約、本文、写真・図等の文書構成要素データを、1つの記事という単位の中のデータとして管理する。上記追加手段は、記事に優先度情報を追加する。上記枠表示手段および上記割付け手段は、上記追加手段により追加された優先度情報の順に、記事に含まれる各文書構成要素データについて、連続して処理を行う。

[作用]

上記枠表示手段により、ユーザが指定した文書構成要素データの大きさを視覚的に認識でき、上記割付け手段により、領域相互の配置を細かく意識することなく、文書のレイアウト作業を行うことができる。

上記枠形状変更手段により、文章が無駄なく入る形状を、画面上で確認しながら、上記枠の変形

を行なえ、段移動手段は、段単位でレイアウトを決定する文書において、素早く正確な配置決定を行なうことができる。

また、上記割付け手段は、上記枠に対応する領域に重なる既存領域の範囲を、領域同士が重ならないように、該既存領域の形状を変更するので、ユーザが指定した通りに文書構成要素データを割付けることができる。

また、上記割付け手段は、既存領域に重なる上記枠に対応する領域の範囲を、領域同士が重ならないように、該枠に対応する領域の形状を変更するので、すでにレイアウトを決めた既存領域の形状を変えずに、文書のレイアウトを順次行なうことができる。

さらに、本発明は、上記管理手段により、関連して割付けるべき見出し、要約、本文、写真・図等の文書構成要素データを、ひとまとめの記事の中のデータとして管理し、文書を記事の集りとして管理するので、ユーザが記事を指定することにより、ユーザが指定した記事に含まれる各文書

モリ、4は表示メモリ、5は表示制御装置、6は表示装置、7は磁気ディスク、8は磁気ディスク制御装置、9はキーボード、10は画面を指示する入力デバイスであるマウス、11は入力デバイス制御装置、12はプリンタ、13はプリンタ制御装置を示し、1はそれらを統合する共通バスである。

表示メモリ4は、表示すべき画面情報を格納し、表示制御装置5は、周期的に表示メモリ4の情報を読み出して、それらを映像信号に変換し、表示装置6に送出する、これにより、表示装置6には、常に、表示メモリ4の画面情報を反映した表示が行われる。

主メモリ3には、本実施例に係るプログラムおよびデータが格納され、演算処理装置2が主メモリ3上のプログラムおよびデータを解釈し、各制御装置に処理を要求することにより、文書の入力、表示、印刷や磁気ディスク7への登録等の文書処理を行う。

第4図は、本文書処理システムが編集対象とする文書の頁の例を示したものである。

構成要素データについて、上記枠表示手段および上記割付け手段は、連続して処理を行う。従って、記事を単位として、その記事に関係する文書構成要素データをまとめて割付け処理でき、記事レイアウトの操作性を改善できる。

また、上記追加手段により、記事に優先度情報を追加し、上記枠表示手段および上記割付け手段は、上記追加手段により追加された優先度情報の順に、各記事に含まれる文書構成要素データについて、順次、処理を行う。従って、ユーザが重要なと考える記事を良い位置に優先的にレイアウトでき、新聞などのレイアウト決定の上で操作性向上を図ることができる。

[実施例]

以下、本発明の一実施例を図面を用いて説明する。

第3図は、本発明の一実施例の文書レイアウト編集装置を適用する文書処理システムのハードウェアの一例である。

第3図において、2は演算処理装置、3は主メ

頁の用紙20から周囲の余白24を除いた部分を文書エリア25と呼び、この範囲に文書データが配置される。第4図の例は、縦書きの文書であり、複数の段23から成っており、用紙20上方の段23より順に、段番号1、2、3、4…が割り振られ、各段23は、段番号で一意に識別することができる。

ここで、文章は、第4図に示すように、段23を単位として右側より縦書きで流し込まれ、いくつかの段23の一部を組合せた領域22(図中の実線で囲まれた多角形の範囲)に、ひとまとめの文章データを格納することができる。領域22には、段23をまたいで配置する見出しのデータや、图形、写真のデータを格納することもできる。29は、まだデータが配置されていない空白領域である。

第4図は縦書きの文書の例であるが、横書き文書の場合は、段23が用紙20左側から順に横方向に並び、文章は、各段23を単位として、用紙20上方から横書きに流し込まれる点が異なる。文章を流し込む領域22の定義は、縦書きの場合と全く同じである。

以上のような頁が複数枚集まって構成された文書を作成する文書レイアウト端末装置の処理について、以下に説明する。

まず、以上の文書構造を実現するためのテーブル構造について、第5図および第6図を用いて説明する。

直に割付ける文書の構成要素データは、磁気ディスク7上ではファイルと呼ばれる形式で格納され、これらのデータを一意に識別できるよう、ファイル名称を定義する。文書レイアウトを処理するプログラムは、このファイル名称を指定することにより、磁気ディスク7から対応する構成要素データを読み出すことができる。

第5図の90～98は、磁気ディスク7上に格納しておく文書の構成要素データの形式を示している

一般的に、新聞等の文書では、論理的な構造として「記事」という単位が考えられ、1つの記事に対して、それに関係する見出し、要約、本文、図、写真などの構成要素が考えられる。

本実施例では、各構成要素のデータは、それぞ

このデータに基づき、頁への文字列の割付けを決定する。

このような記事のデータを文書に割付けるための管理テーブルを第6図(4)に示す。

記事中のデータを頁に割付けるときには、記事ファイル90がさし示す見出し、要約、本文、図・写真等の個々のデータを、領域22に直接対応させて、領域22内に、それらのデータを配置する。

領域22は、領域管理リスト30により管理され、次のような構成をとる。

領域数テーブル31は、文書に設定されている領域22の数を格納し、領域ポインタ・テーブル35は、領域数テーブル31が示す数の書式テーブルポインタ36および段テーブルポインタ37を持つ。領域22は、0から始まる領域番号で識別する。また、領域番号は、データの割付けを行うとき、どちらの領域22を優先するかを示す優先度の意味付けも持っており、番号が若い領域22ほど優先度が高いものとする。

書式テーブル40は、その領域22に入るデータの

れ独立したファイルとして磁気ディスク7に格納され、記事に対応して設ける記事ファイル中に、関係する構成要素のファイル名称を格納することにより、記事から、その構成要素を簡単に参照できる構造とする。

具体的には、記事ファイル90は、各記事に含まれるデータをさし示す見出しファイル名称91、要約ファイル名称92、本文ファイル名称93、図・写真ファイル名称94といったデータを持つ。対応するデータがない場合は、ファイル名称には、NULLコードが格納される。

また、見出しファイル95には、見出し用文字列の文字コードが、要約ファイル96には、要約の文章の文字コードが、本文ファイル97には、本文の文字コードが、図・写真ファイル98には、図や写真を2値化したイメージデータが、それぞれ、格納されている。

なお、本実施例においては、見出し、要約、本文といったデータの種別ごとに、標準的に用いられる書式を定めた書式データ99が用意されており、

種類（本文、図・写真、見出しなど）を示す領域属性41のフィールド、領域22内が文字列のデータの場合に有効な文字間隔42、行間隔43、文字サイズ44、その領域22で標準として用いられる書体や文字の飾りなどの情報である文字属性45のフィールド、および、その領域22に流し込まれるデータのファイル名46のフィールドを保持する。

領域属性41が本文であれば、その領域22の割付け時に、領域形状の変更が可能とする。他の属性ならば、設定した領域形状をくずさずに割付けを行う。

一般的に、新聞のような文書では、本文、見出し、要約等の構成要素データの種類に応じて、その書式は決まっていることが多い。そのため、領域22が新たに作成されたときには、そのデータの種類に応じて、あらかじめ定められた書式データ99の内容に従って、書式テーブル40を生成する。

ただし、書式テーブル40自体は、領域22ごとに持つため、後から領域22単位で書式を変更することは、自由に行うことができる。

つまり、領域22を、各段23ごとに文章を割付け
る矩形範囲の集合として考えることができる。

このため、本実施例では、各矩形範囲に対応し
て、その位置や大きさを管理する段テーブル50を
設け、領域22を段テーブル50を連結したリストと
して管理する。

段テーブル50は、段テーブル50を結合するポイ
ンタ51、その割付け範囲が存在する頁の番号52と
その中の段23の番号53、その段23内で割付ける範
囲を示す開始点座標54および終了点座標55のフィ
ールドを持ち、また、その範囲内に書式テーブル
40で定義された文字列が入りうる文字数を示す格
納文字数56のフィールドを持つ。

段テーブル50のリストは、頁番号52および段番
号53の若いものの順に接続され、リスト末尾のポ
インタ51には、NULコードが格納されて、末
尾であることを意識できるようにする。開始点座
標54および終了点座標55は、文書エリア25の原点
から見た単位の段内の範囲を示し、例えば、第
4図の縦書き文書の場合なら、文書エリア25の原

ブルを生成する必要が生じたときには、ロック
ブルから獲得され、削除時には、ロックブル
に返却される。

ロックブルは、第6図(b)に示すようなブロ
ックブルリスト500で管理され、ブルポイン
タ501からポインタ503で、共通の形式のブロ
ック502を連結しており、ブロック502を獲得する
ときは、ブルポインタ501の示すリストの先頭
から取り、返却するときも、先頭に挿入する。
なお、ロックブルは、割付け処理に必要な十
分な量を用意する。

頁書式テーブル110は、各頁の書式情報を持つ
ものであり、第6図(a)に示すように、頁数111の
フィールドと、各頁について、頁サイズ(縦、横
の長さ:単位)112、上下左右の余白値(単位)
113、段組数114、段と段の間隔(単位)
115を保持している。

次に、以上のテーブルを用いて行う文書レイア
ウト処理について説明する。

第1図は、本実施例で本文データを割付ける際

点(右上端)を基準として、段23内の領域22の仕
切り線の横方向座標値を格納する。

また、まだ割付けが行われていない部分は、各
段に存在する矩形の空エリアを示す空エリアテー
ブル65を連結した空エリヤリスト60として管
理する。

空エリアテーブル65は、段テーブル50と同じ構
成をしているが、格納文字数56のフィールドに相
当するフィールドは無効である。空エリアテーブ
ル65は、空エリアポインタ61を先頭に、頁番号67
および段番号68の若い順に、ポインタ66で連結さ
れる。

文書にまだ割付けられていない状態では、各段
23について、段23全体の大きさを持つ空エリアテー
ブル65が空エリヤリスト60に接続されており、
領域22を割付けることにより、空エリアテーブル
65が削除されたり、範囲が細分化された空エリヤ
テーブル65が段内に複数生成される。

以上の段テーブル50および空エリヤテーブル65
は、共通のロック形式をしており、新たにテー

の画面表示と、その処理概要フローチャートを示
したものである。

まず、ステップ201で、第1図(a)に示すように、
割付ける記事名の一覧を画面に表示する。ユーザ
が、マウス10により、目的とする記事名上にカーソ
ル27を動かして、マウス10に付いているボタン
を押すと、ステップ202で、その記事が選択され
る。

以後、このような操作を「マウス10で記事名を
指定する」というように記述することにする。

ユーザにより記事が指定されると、ステップ203
で、第1図(b)に示すように、記事を構成するデー
タファイル名を画面に表示する。ステップ204で、
ユーザが、目的とするファイル名をマウス10で指
定すると、ステップ205で、第1図(c)に示すよう
に、頁のレイアウトを表示する。

この例では、#0、#1、#2の3つの領域22
がすでに設定されており、#0には見出し、#1、
#2には本文が割付けられている。

ステップ220では、その上に、第1図(c)の26-

1に示すような矩形の太線で、仮枠26を表示する仮枠表示処理を行う。この仮枠表示処理は、第7図に示した処理フローチャートに従って行われる。

第7図において、仮枠表示処理では、まず、ステップ229で、空エリアポインタ61がNULコードかどうか調べ、NULコードならば、空エリアがなく割付けが不能のため、ステップ401で、エラーメッセージを表示し、処理を終了する。

空エリアポインタ61がNULコードでなければ、ステップ221以下の処理を行う。まず、ステップ221で、表示している頁の頁書式テーブル110を参照し、縦方向の頁サイズ112から上下左右の余白値113およびすべての段間の幅を引き、これを段組数114で割ることにより、1段の段幅を求める。

続いて、ステップ222で、この値を、本文の書式データ99によって定義される縦方向の文字サイズ44と文字間隔42を加えたもので割り、その整数部をとることにより、1行に入りうる文字数nを求めることができる。

と段数 μ で割り、その整数単位で切上げた値が、求める各段の本文行数Mとなる。

続いて、ステップ226で、このMの値に、横方向の文字サイズ44と行間隔43を加えたものを掛けることにより、仮枠26の横幅が求まる。また、仮枠26の縦幅は、 μ 段分の段幅と、その間の段間隔($\mu - 1$)個分とを加えることにより求まる。

最後に、ステップ227で、文書エリア25の右上端原点から、求めた仮枠26の横幅および縦幅分の矩形を、第1図(c)の26-1に示すように表示し、ステップ228で、仮枠テーブル130に、枠開始点座標131には縦も横も0の値を、枠横幅132および枠縦幅133にはステップ226で求めた値を、段数134には μ の値を、先頭段135には1の値を、それぞれ、設定する。

次に、第1図に戻って、画面指示デバイスであるマウス10を用いた枠形状変更操作(ステップ240)を行い、仮枠26を、第1図(c)の26-2に示すように、段数および位置をユーザの選むように設定する。

次に、なるべく少ない段数で本文が流し込めるようにするために、表示する仮枠26が最低何段必要かを、ステップ223～ステップ224で求める。

このためには、まず、ステップ223で、横方向の頁サイズ112から左右の余白を除いたものを、本文の書式データ99の横方向の文字サイズ44と行間隔43を加えたもので割り、その整数部をとることにより、1段に入りうる行数mを求める。

さらに、ステップ224で、磁気ディスク7から読み出した本文データを参照して、その文字数Nを求め、これを、 $n * m$ (1段に入る文字数)で割り、その整数部+1を求めることにより、本文格納に必要最小限の段数 μ を求めることができる。

ただし、本文データの文字数Nは、データ中に改行文字があった場合、そこから次の行末まで文字が詰まっているものとして、その分の文字数を加算して求めておくものとする。

次に、ステップ225で、段数 μ の各段に同じ幅だけ文字を流し込んだときに、何行必要かを求める。このために、本文の文字数Nを1行の文字数

この枠形状変更操作を説明するための画面表示例を第8図に示し、処理フローチャートを第9図に示す。

第9図(a)はメインフローチャートであり、ステップ241で、ユーザが、マウス10で領域枠を指定すると、ステップ242で、その指示位置に相当する文書エリア25右上端からの座標値(X, Y)を取得し、それが仮枠26の上辺の近傍であれば、上辺移動処理(ステップ260)を行う。上辺移動処理は、上辺を移動することにより、仮枠26の形状を変更するものである。本フローチャートにおいて、 ϵ は、近傍を示す幅の値を示している。

上辺の近傍でなければ、ステップ243で、下辺の近傍かどうかを判定し、下辺の近傍でなければ、ステップ244で、右辺の近傍かどうかを判定する。右辺の近傍でもなければ、ステップ255で、左辺の近傍かどうかを判定し、左辺の近傍でもなければ、ステップ246で、仮枠26の内部かどうかを判定する。

マウス10の指示点が下辺、右辺、左辺のいずれ

かであれば、それぞれ、下辺移動処理（ステップ270）、右辺移動処理（ステップ280）、左辺移動処理（ステップ290）を行う。

また、指示点が仮枠26内部であれば、仮枠26全体を移動する枠移動処理（ステップ250）を行い、仮枠26外部であれば、何もせずに、再びマウス10による指示待ち（ステップ241）に戻る。

ここで、第8図の操作例を説明するために、枠移動処理および下辺移動処理について、特に詳しく説明する。

第9図(b)は枠移動処理の処理フロー チャートである。

枠移動処理は、ステップ251で、マウス入力を受付けると、マウスボタンを押している間、ステップ253以下の処理を行い、マウスボタンが離されると、処理を終了する。

マウスボタンが押されていると、まず、ステップ253で、その時のマウス10の移動量を取得し、ステップ254で、仮枠テーブル130の内容に従って、表示した仮枠26を一旦消去する。次に、ステ

ップ259-aで、枠始点座標131のx(横方向の値)にマウス10の移動量のX(横方向)を加え、その値がマイナス、すなわち、仮枠26が右側にはみ出するようならば(ステップ259-b)、ステップ259-cで、枠始点座標131を0にして、仮枠26の位置を右端に調整する。また、同様に、仮枠26が左側にはみ出するようならば(ステップ259-d)、ステップ259-eで、仮枠26の位置を左端に調整する。

このような処理により、マウス10を動かすことで、横方向にはなめらかに、縦方向には段単位に、仮枠26全体の移動を行うことができ、例えば、第8図(a)の状態で、カーソル27を仮枠26内部に置いて、マウスボタンを押したまま右下に動かすと、第8図(d)のように仮枠26を動かすことができる。

第9図(c)は下辺移動処理の処理フロー チャートである。

この処理でも、まず、ステップ271で、マウス入力を受付け、マウスボタンが押されている間、ステップ272以下の処理を行い、マウスボタンが離されると、処理を終了する。

マウスボタンが押されていると、まず、ステップ272で、マウス10の移動量を取得し、ステップ273で、一旦仮枠26を消去する。

次に、ステップ274で、マウス10の縦方向の移

動量Yを判定し、プラスでかつ表示していた仮枠26の段数134が真内の段組数に等しくなければ(ステップ275)、ステップ276で、段数134を増やし、ステップ278で、段を増やしたときに、各段に格納する本文の行数Mを求め、ステップ279-aで、これを元に、枠横幅132および枠縦幅133を求めて、仮枠テーブル130に設定し、ステップ279-bで、仮枠26を表示し直した後、再びマウス入力を受付ける(ステップ251)。

ステップ256でマウス移動量Yがマイナスでか

動量Yを判定し、プラスでかつ表示していた仮枠26の段数134が真内の段組数に等しくなければ(ステップ275)、ステップ276で、段数134を増やし、ステップ278で、段を増やしたときに、各段に格納する本文の行数Mを求め、ステップ279-aで、これを元に、枠横幅132および枠縦幅133を求めて、仮枠テーブル130に設定し、ステップ279-bで、仮枠26を表示し直した後、再びマウス入力を受付ける(ステップ251)。

ステップ274の判定でマウス10の縦方向の移動量Yがマイナスであれば、ステップ220で求めた本文を格納するのに必要最小限の段数mと仮枠26の段数134とが一致しなければ(ステップ277-a)、ステップ277-bで、段数134を1つ減らし、ステップ278～ステップ279-bの処理を行い、新たな形状の仮枠26を表示する。

また、マウス10の縦方向の移動量Yが0であったり、段数134がとりうる最大のときにマウス10を下に動かしたり、段数134がとりうる最小のときにマウス10を上に動かしたりした場合は、形状

は変更せず、ステップ 279 - b で、仮枠 26 を表示し直す。

以上の処理により、第 8 図(a)の状態から、仮枠 26 の下辺をマウス 10 で指示し、マウスボタンを押したままマウス 10 を下に動かすことにより、第 8 図(b)、第 8 図(c)のように、順次、仮枠 26 の段数が変化すると共に、本文のデータ量に合せて、各段の横幅が調整される。

このような枠形状変更操作（第 1 図のステップ 240）により、第 1 図(c)の 26 - 2 のように、仮枠 26 を変形・移動した後、ステップ 208 で、キーによる割付け実行指示を行うことにより、第 1 図(d)に示すように、仮枠 26 を正式な領域 22 - 3 として設定し、その中に本文を流し込むと共に、ステップ 300 で、仮枠 26 に重なっていた #1, #2 の領域 22 の部分を空領域に割当て直し、ステップ 210 で、その結果を画面に表示する。

このステップ 300 の割付け処理の処理内容を、第 10 図の処理フロー チャートに示す。

この処理においては、まず、ステップ 310 で、

26 内にあった空白領域 29 についても、空エリアテーブル 65 から削除する。

最後に、保留しておいた領域部分を、順次、空白領域 29 に割付けて（ステップ 420 ～ステップ 480）、保留しておいた領域 22 がなくなれば（ステップ 470）、レイアウトが決定したことになる。

本実施例では、割付け処理で、仮枠 26 と重なった領域部分を、一旦保留しておくため、第 11 図に示すような保留テーブル 145 を連結し、保留ポインタ 141 から始まるリストとして保留リスト 140 を設ける。

保留テーブル 145 は、段テーブル 50 と同じブロックテーブルから獲得したブロックを用いたワークテーブルであり、段テーブル 50 で格納文字数 56 のフィールドが、所属領域番号 150 のフィールドに変わっている点が異なり、このフィールドにより、保留されているデータがどの領域 22 に属していたものかを識別する。

また、保留テーブル 145 は、所属領域番号 150 で昇順にソートされ、所属領域が同じテーブルの

流し込むデータの種類ごとに定められた書式データ 99 に従って、書式テーブル 40 を作成する。すなわち、文字間隔 42、行間隔 43、文字サイズ 44、文字属性 45 を複写し、領域属性 41 には、データの種類を、データファイル名 46 には、流し込むデータのファイル名を設定し、書式テーブル 40 を作成する。

次に、ステップ 320 で、領域数テーブル 31 の値を 1 つ増やして、領域ポインタ・テーブル 35 の末尾フィールドの書式テーブルポインタ 36 に、作成した書式テーブル 40 の先頭アドレスを設定し、段テーブルポインタ 37 に NUL コードを設定する。

続いて、仮枠 26 の範囲を段単位に分割して段テーブル 50 を作成し、正式な領域 22 として登録する。

このため、まず、仮枠 26 の先頭の段から順次、各段について（ステップ 340、410、415）、ステップ 350 で、仮枠 26 の範囲を正式な領域 22 として登録し、ステップ 370 で、その領域 22 に重なってしまった他領域 22 の範囲を保留しておいて、段テーブル 50 からは削除し、ステップ 390 で、仮枠

間では、段番号 146 で昇順にソートされ、段番号 146 の等しい場合には、開始点座標 148 で昇順にソートされて連結されている。

次に、このようなテーブルを使用した割付け処理の詳しい処理内容を説明する。

まず、ステップ 350 で、仮枠 26 を段単位で分割した部分を領域 22 として登録する処理を行う。

このステップ 350 の処理を、第 12 図を用いて説明する。

この処理では、ステップ 351 で、新たな段テーブル 50 のエリアを 1 つ確保して、ステップ 352 で、その貯番号 52 に、表示中の貯番号を設定し、ステップ 353 で、段番号 53 に、処理対象としている段の番号を設定し、ステップ 354 で、開始点座標 54 に、仮枠テーブル 130 の枠始点座標 131 の X を設定し、ステップ 355 で、終了点座標 55 に、枠始点座標 131 の X に枠横幅 132 の X ピクセルを設定し、ステップ 356 で、格納文字数 56 に、ステップ 220 で求めた（1 行文字数 N）*（各段の行数 M）の値を設定し、ステップ 357 で、ポインタ 51 に、NUL

レコードを設定して、新たな段テーブル50を作成する。

統いて、ステップ358で、段テーブルポイント37を判定し、NULレコードならば、ステップ361で、作成した段テーブル50を段テーブルポイント37に接続し、段テーブルポイント37がNULレコードでないならば、ステップ359で、段テーブル50をたどって、リストの末尾の段テーブル50を見つけ、ステップ360で、作成した段テーブル50をこれに接続する。

これにより、頁、段の順に、整列された段テーブル50のリストが作成され、ステップ350の処理を、仮枠26が存在する各段で繰返すことにより、仮枠26の範囲に対応する領域管理リスト30が作成される。

次に、第10図のステップ370では、仮枠26を段単位で分割した範囲に重なる他領域22を求めて、その段テーブル50を削除し、その部分のデータを、一旦保留するために、保留テーブル145を作成し、保留リスト140に接続する。

形状を変形するべきではないため、ステップ386で、エラー表示を行い、割付け処理を強制終了させる。

新たな領域22に亘る領域22の部分の段テーブル50が見つかった場合は、その既存の段テーブル50の範囲が新たな段テーブル50の範囲に含まれていれば（ステップ376）、ステップ379で、見つけた既存の段テーブル50を段テーブル50のリストからはずし、そのままの形で保留テーブル145とし、ステップ381で、保留リスト140に挿入する。

亘っている既存の段テーブル50の範囲に、新たな段テーブル50がすべて含まれている場合は（ステップ377）、亘った範囲の保留テーブル145を新たに作成し、ステップ380で、亘なり部分を除き、2つに分れた部分について、段テーブル50をもう1つ追加して作成し、ステップ381で、作成した保留テーブル145を保留リスト140に挿入する。

ステップ377の判定で、亘っている既存の段テーブル50の範囲に、新たな段テーブル50の一部

このステップ370の処理について、第13図を用いて説明する。

ここでは、#0の領域22から新たに設定した領域22の1つ前の領域22までのすべての領域22について（ステップ371、383、384）、新たに設定した領域22の各段での範囲に亘る他領域22の部分を求める（ステップ372～ステップ382）。

このため、対象となる領域22の段テーブル50の頁番号52および段番号53が、新たに加えた領域22の段テーブル50の頁番号52および段番号53と一致し（ステップ374）、対象となる領域22の段テーブル50の開始点座標54および終了点座標55と、新たに加えた領域22の段テーブル50の開始点座標54および終了点座標55とを比べて、両方の領域22の少なくとも一部が重なっており（ステップ375）、かつ、亘った領域22が本文ならば（ステップ385）、ステップ386以下の保留テーブル作成処理を行う。

ステップ385の判定で、本文以外の領域22が亘っているときには、見出しや図・写真等の領域

でも含まれていない場合は、新たな段テーブル50と亘っている既存の段テーブル50とは、互いに部分的に重なっている。このときには、ステップ378で、亘る範囲の保留テーブル145を作成し、亘っている既存の段テーブル50から亘なり部分を除くよう開始点座標54および終了点座標55を変更し、ステップ381で、作成した保留テーブル145を保留リスト140に挿入する。

ステップ381における挿入処理は、保留リスト140が所属領域番号150、頁番号および段番号147、開始点座標148の優先順で昇順にソートして挿入しておき、後で保留された範囲を頁に割付ける処理を行いやくする。

この次に、第10図のステップ390で、仮枠26内にあった空白領域29が空エリアでなくなったため、その範囲を空エリヤリスト60から削除する処理を行う。

このステップ390の処理を、第14図を用いて説明する。

まず、空エリアを先頭から順次参照し（ステッ

ア 392, 400)、対象となる空エリアの空エリアテーブル65の頁番号67および段番号68と、新たな段テーブル50の頁番号52および段番号53とが一致し(ステップ393)、かつ、対象となる空エリアの空エリアテーブル65の開始点座標69および終了点座標70の範囲と、新たな段テーブル50の開始点座標54および終了点座標55の範囲とが重なる空エリアを求める(ステップ394)。

そのような空エリアが見つかったとき、新たな段テーブル50に空エリアテーブル65の範囲がすべて含まれていれば(ステップ395)、ステップ398で、その空エリアテーブル65を空エリアイスト60から削除し、新たな段テーブル50の範囲が空エリアテーブル65に含まれてしまうときには(ステップ396)、ステップ399で、空エリアテーブル65から重なり部分を除いた2つの部分について、空エリアテーブル65をもう1つ追加して、空エリアテーブル65を分割生成する。

また、ステップ396の判定で、新たに加えた段テーブル50の一部でも、空エリアテーブル65に含

する領域22が存在する最上段から最下段まで(ステップ431, 441, 442)、ステップ432またはステップ433で、領域22の右または左に隣接する空エリアを、空エリアイスト60を調べて求め、ステップ434で、そこに入りうる文字数を書式テーブル40から求め、求めた文字数が保留テーブル145より格納文字数が大きければ、ステップ437で、空エリアテーブル65の開始点座標69および終了点座標70を保留テーブル145の大きさ分狭め、逆に、空エリアテーブル145の方が小さければ、ステップ436で、その空エリアテーブル65全体を空エリアイスト60から削除する。

いずれの場合も、ステップ438で、空エリアテーブル65を減らした範囲について、保留テーブル145の所轄領域番号150に対応する段テーブル50のリストに追加し、ステップ439で、保留テーブル145の範囲も新たに段テーブル50が拡大した分だけ減らす。

これにより、保留テーブル145の範囲がなくなってしまった場合には(ステップ440)、第10回

まれていない場合は、新たな段テーブル50と空エリアテーブル65とが部分的に重なっている場合であり、この場合には、ステップ397で、空エリアテーブル65の開始点座標69および終了点座標70から、新たな段テーブル50と重なる範囲分を狭める。

次に、第10回のステップ430で、残っている保留リスト140につながれた保留テーブル145を割付けてなくす処理を行う。

このステップ430の処理を、第15回を用いて説明する。

この割付け処理では、仮枠26に重なった部分がある領域22の一一番上の段23から、その領域22に隣接する空白領域29を順次割付け、それでも足りなければ、領域22のある範囲より下の段23に割付けている。

ただし、このような割付け方法は、作成する文書の種類によっては、組版ルールが異なるため、1つの最も良い方式が決められる性質のものではなく、他の様々な割付け方法が考えられる。

ここで示した例では、保留テーブル145の所轄

のステップ460に戻って、該保留テーブル145を保留リスト140から削除する。

保留テーブル145の領域22が存在する段23で、保留テーブル145をなくすことができない場合は(ステップ441)、その下の段23から順に(ステップ443, 449)、空エリアがあれば(ステップ446)、保留テーブル145の割付け処理を、ステップ435～ステップ439と同様に行い(ステップ447)、保留テーブル145がなくなれば(ステップ448)、第10回のステップ460に戻って、該保留テーブル145を保留リスト140から削除する。

ただし、割付けたときに、そのエリアが別の頁になってしまったときには(ステップ444)、本処理方式では、ステップ445で、対象テーブルの範囲を所定文字数だけ狭やし、余分に領域22を広げている。これは、文章を頁にまたがって割付けるときには、頁末に「次頁に続く」等のメッセージを附加することが組版では一般的であり、そのメッセージを入れるだけの余分の領域22が自動的に確保される。

また、保留テーブル 145 の属する領域 22 より下に連続して空エリアがなければ(ステップ 446)、ステップ 450 で、割付けができない旨のメッセージを表示して、割付け処理を強制終了する。これは、一般的な組版ルールでは、段 23 を 1 つ以上あけて 1 つの文章を流し込むのは、良くないとされているためである。

これらの処理で、エラーが起こった場合は、再び、仮枠 26 を表示する。ユーザは、仮枠 26 の形状や位置を設定し直して、割付けの実行を指示し直す。

以上の処理により、ユーザが、記事のデータファイルをマウス 10 で指示することにより、そのデータ量に応じた仮枠 26 が表示され、ユーザが仮枠 26 の形状や位置をマウス 10 で指示して、割付け実行を指示するだけで、仮枠 26 通りにデータを割付けると共に、仮枠 26 に重なるすでに設定された領域を自動的に割付けるため、組版のルールを知らない初心者ユーザでも、簡単に高度な割付け作業を行うことができる。

次に、仮枠 26 と他の領域とで重なる部分を求めるが、本実施例では、ステップ 340 で、仮枠 26 から生成した方の段テーブル 50 から、重なり部分を削除し、その部分の保留テーブル 145 を作成して保留リスト 140 に追加する。この後、ステップ 370 で、保留リスト 140 の解消を第 10 図のステップ 420～ステップ 480 と同様に行うことにより、新たに仮枠 26 で設定した領域で、他領域と重なる部分が空白領域に割当てられ、変形されることになる。

最後に、形状が決定された領域は、すべて空白領域だった部分に割付けられるため、ステップ 380 で、割付けたエリアすべてを、空エリヤリスト 60 中の空エリヤテーブル 65 から削除し、処理が完了する。

この方式では、自分が設定したい場所と異なる場所に割付けが行われる可能性が多いが、すでに設定された領域の変形を行わないため、一般に、重要な記事から順に記事を割付けることを考えると、先に示した方式に比べて、レイアウトの決定

割付け処理の別の実施例として、第 16 図に示すような割付け方式がある。

第 16 図(a)に示すように、+0 の領域が割付けられた状態から、第 16 図(b)に示すように、仮枠 26 を設定し、割付けの実行を指示すると、第 16 図(c)に示すように、新たに設定した+1 の領域の方が、すでに割付けられている+0 の領域を避けて変形される。第 16 図(d), (e) は、さらに、+2 の領域を設定した場合の表示例である。

このような方式の割付け処理は、第 17 図に示す処理フローチャートに従って行われる。

すなわち、まず、ステップ 310 で、第 10 図のステップ 310～ステップ 321 と同様に、書式テーブル 40 を作成し、新たに、領域ポインタ・テーブル 35 に登録する。

次に、仮枠 26 のある段の範囲について、上から順に(ステップ 320, 350, 360)、第 10 図のステップ 350 と同様に、その段の仮枠 26 の範囲について、段テーブル 50 を作成し、段テーブル 50 のリストに追加する。

を行いやすくなる。

さらに、第 1 図(a), (b)に示したデータファイルの指定方法とは別に、次のような指定方法が考えられる。

すなわち、ユーザが、第 1 図(a)の記事名一覧で、記事名を指定すると、第 1 図(b)のデータファイル一覧は表示せずに、第 1 図(b)に表示されるはずだった各データファイルのうち、先頭の見出しから、要約・本文・写真・図といった順に、第 1 図(c), (d)の領域の割付け作業を連続して行なえるようにする。

この方式では、ひとまとめの記事単位でその構成要素をレイアウトできるため、新聞のような複数の記事を配置する文書のレイアウトを行う際に、データファイル指定の手間が減り、かつ、ユーザが置く順序でレイアウトを行うことができる。

さらに、別的方式としては、ユーザが割付ける記事名の先頭に、記事の重要度に応じて番号を付けておくと、第 1 図(e)の記事名一覧も表示されず、記事名の番号順に自動的に順次、領域の割付け作

薬を行うことができる。

この方式を用いると、記事の重要度が明確に決まっている文書においては、その重要度の順に割付け作業を行なえ、記事名指定の手間が省ける。

以上説明したように本実施例は、領域決定の前に文章量に応じた仮枠26を表示することにより、従来方式に比べて、文章量に最適な領域割付けを、容易に行なうことができる。

また、仮枠26の変形を、文章量を考慮して、面積一定のまま行なえることにより、ユーザが割付けに必要な範囲を視覚的に確認しながら領域割付けができる。

また、仮枠26で設定される領域とすでに設定された領域とが重ならないよう、自動的に割付けるので、ユーザは、細かな領域の位置決めを考えずに済み、初心者でも容易に文書レイアウトが可能となる。

さらに、記事単位で文書レイアウトが行なえ、記事についても、優先順でレイアウトが行なえるので、ユーザが重要なと思う記事を、構成要素順に、

る記事のファイル構成図、第6図は文書のレイアウトを管理するテーブル構成図、第7図は仮枠表示処理の処理フローチャート、第8図は枠形状変更の画面表示例を示す説明図、第9図は枠形状変更処理の処理フローチャート、第10図は割付け処理の処理フローチャート、第11図は保留リストのテーブル構成図、第12図は新領域の作成処理の処理フローチャート、第13図は保留リスト作成処理の処理フローチャート、第14図は空エリア削除処理の処理フローチャート、第15図は保留リスト削除処理の処理フローチャート、第16図は本発明の第2の実施例の画面表示例を示す説明図、第17図は第2の実施例の処理フローチャートである。

1 … 共通バス	2 … 演算処理装置
3 … 主メモリ	4 … 表示メモリ
5 … 表示制御装置	6 … 表示装置
7 … 磁気ディスク	
8 … 磁気ディスク制御装置	
9 … キーボード	10 … マウス
11 … 入力デバイス制御装置	

従来方式に比べて、操作ステップを少なく実行できる。

[発明の効果]

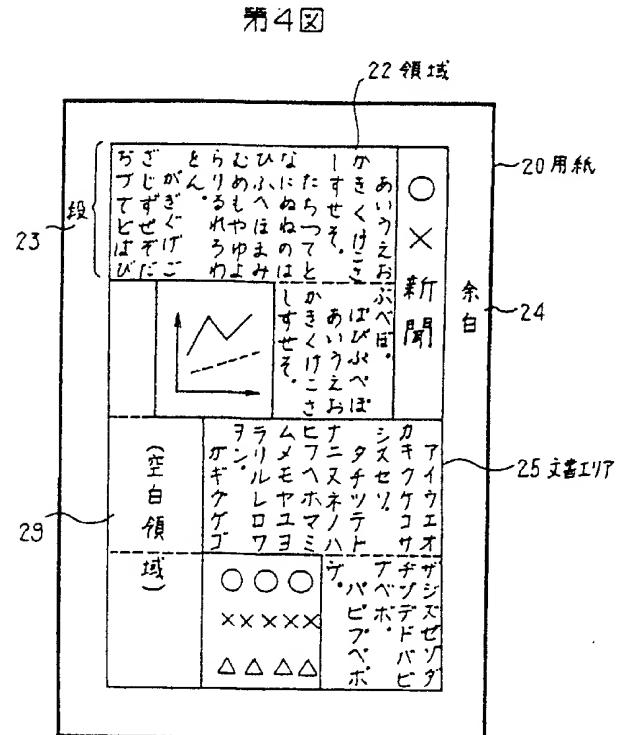
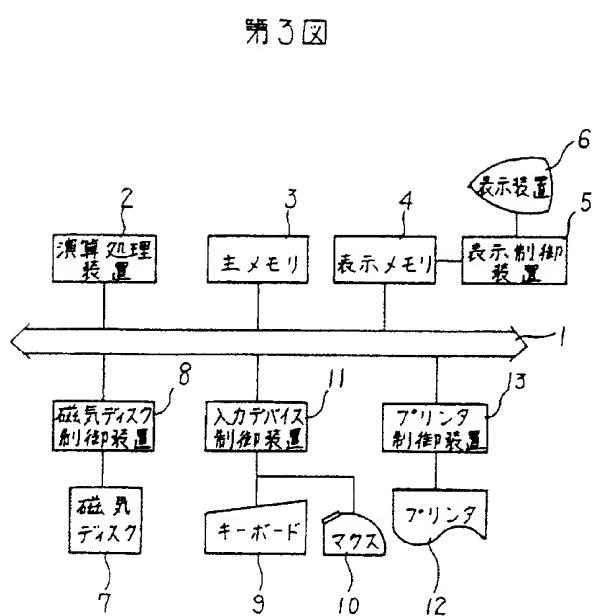
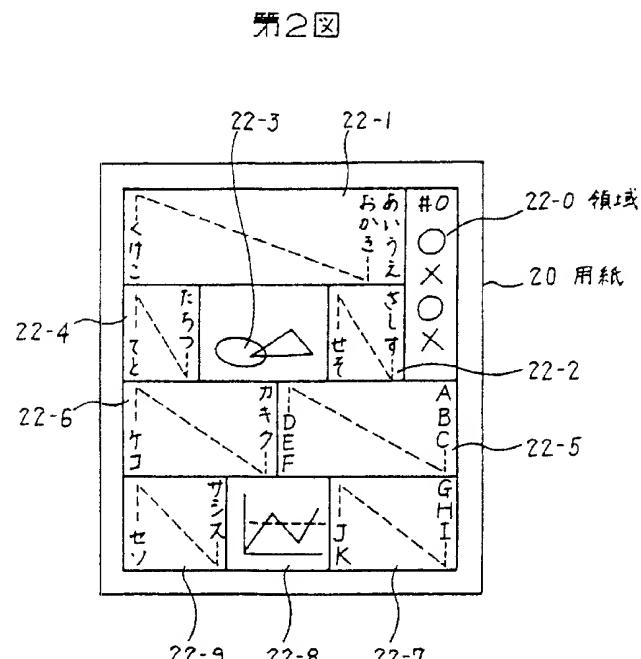
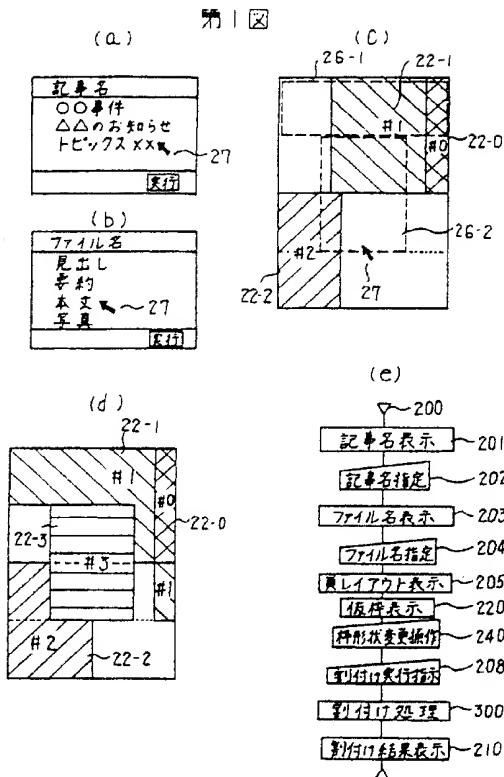
以上説明したように本発明は、組版ルールを知らない初心者に対しても、容易な操作で、ある程度美しい段組み文書を作成することができるという効果がある。

また、多段組み文書の各構成要素データを、内容的につながった単位でまとめて、順次レイアウト可能とするので、レイアウト決定を容易にすることができるという効果もある。

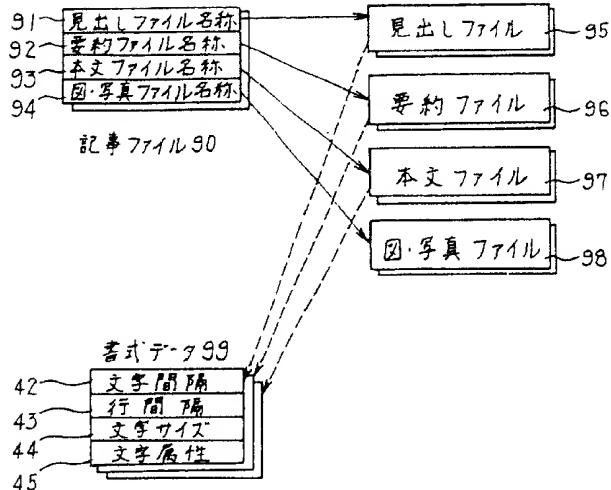
4. 図面の簡単な説明

第1図は本実施例で本文データを割付ける際の画面表示と、その処理概要フローチャートを示した説明図、第2図は従来方式による文書レイアウト例を示す説明図、第3図は本発明の一実施例の文書レイアウト編集装置を適用する文書処理システムのハードウェアの一例を示すブロック図、第4図は本文書処理システムが編集対象とする文書の頁の例を示した説明図、第5図は文書に割付け

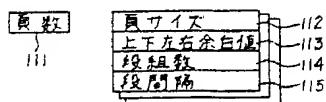
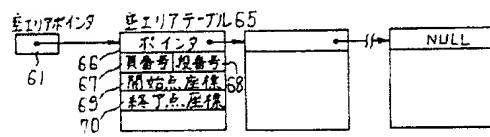
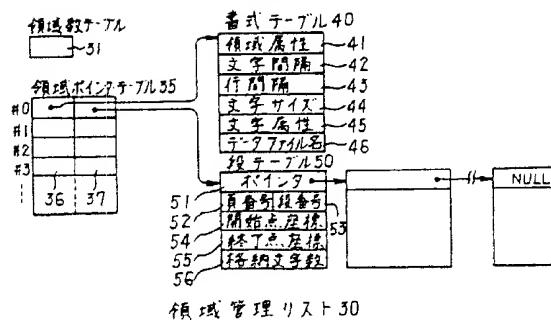
12…プリンタ	13…プリンタ制御装置
20…用紙	22…領域
23…段	24…余白
25…文書エリア	26…仮枠
27…カーソル	29…空白領域
30…領域管理リスト	31…領域数テーブル
35…領域ポインタ・テーブル	
40…書式テーブル	50…段テーブル
60…空エリアリスト	65…空エリアテーブル
90…記事ファイル	95…見出しファイル
96…契約ファイル	97…本文ファイル
98…図・写真ファイル	99…書式データ
110…頁書式テーブル	130…仮枠テーブル
140…保留リスト	145…保留テーブル
500…ロックブルリスト	



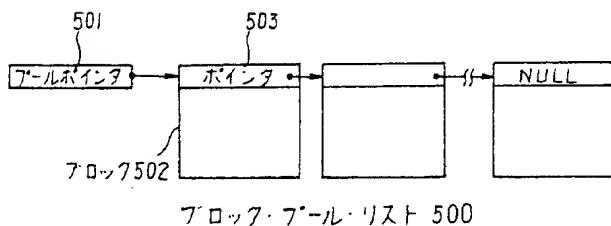
第5図



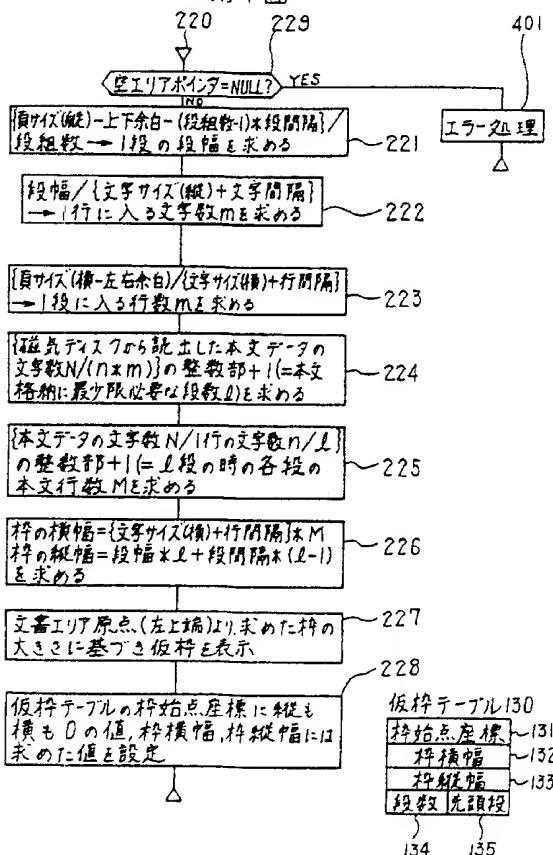
第6図 (a)



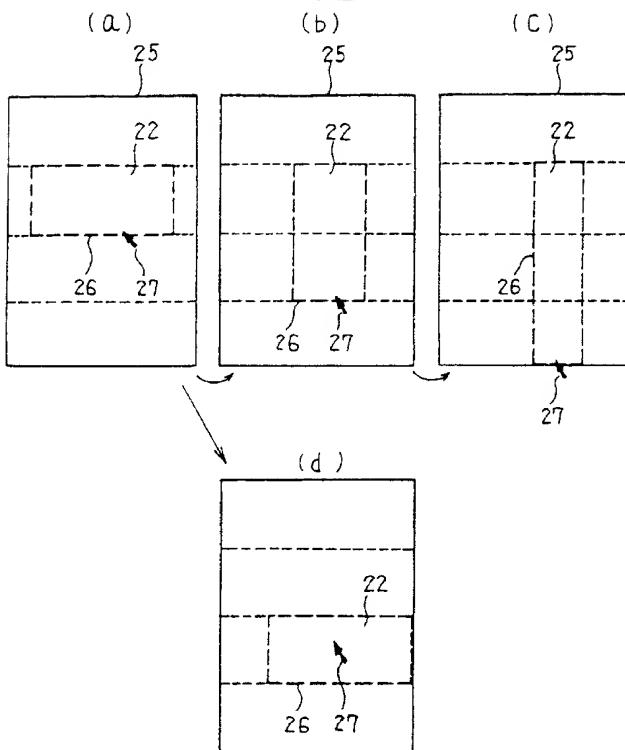
第6図 (b)



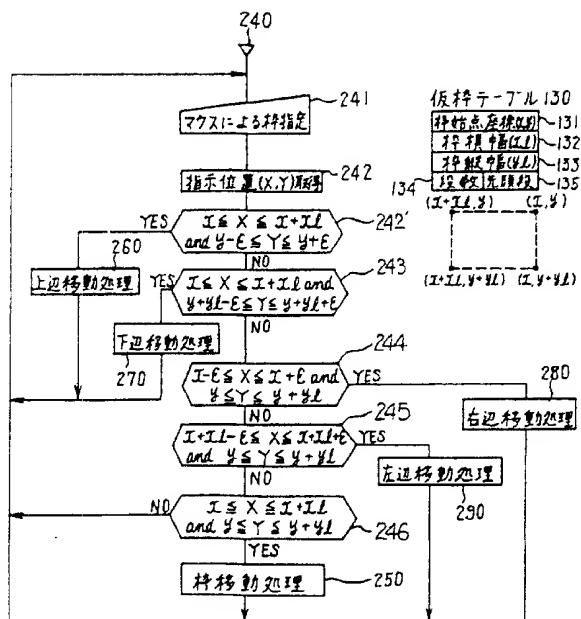
第7図



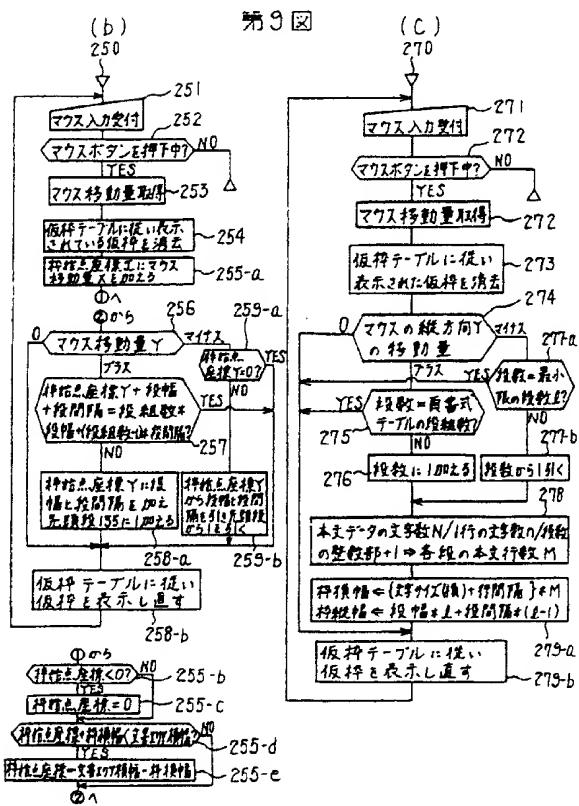
卷八



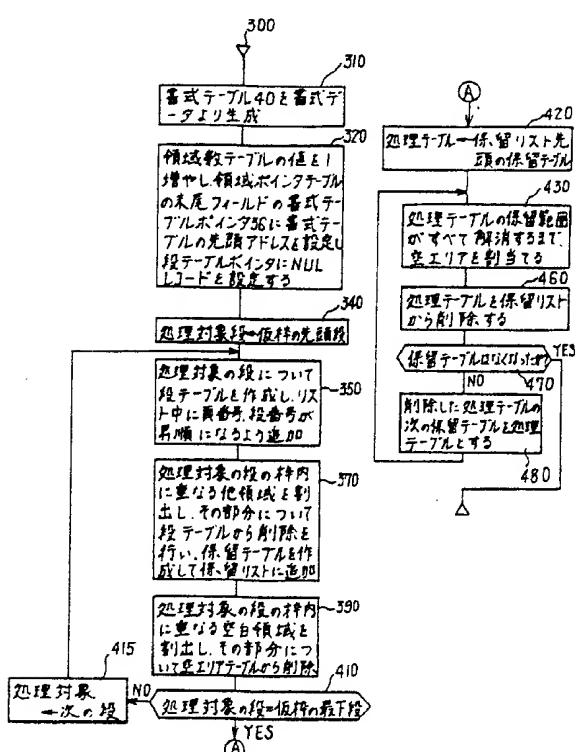
第9図 (a)



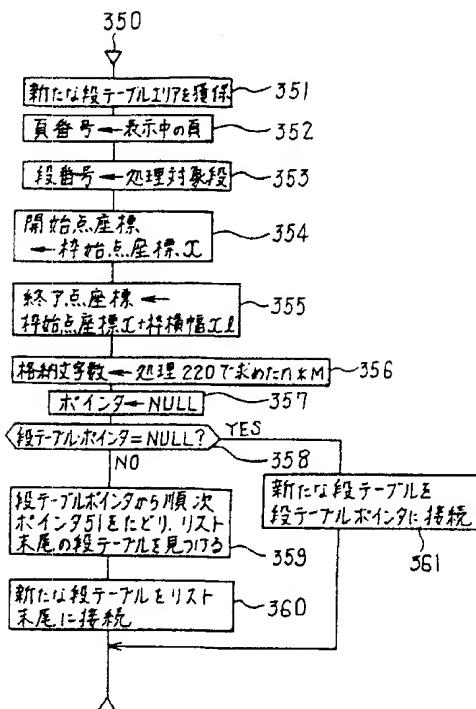
新文獻



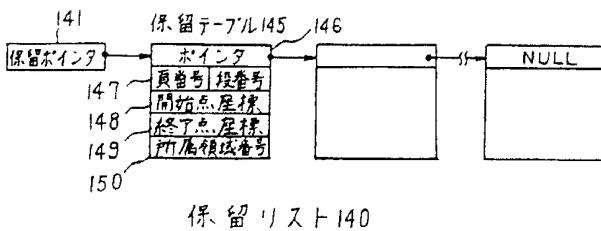
第10回



第12図

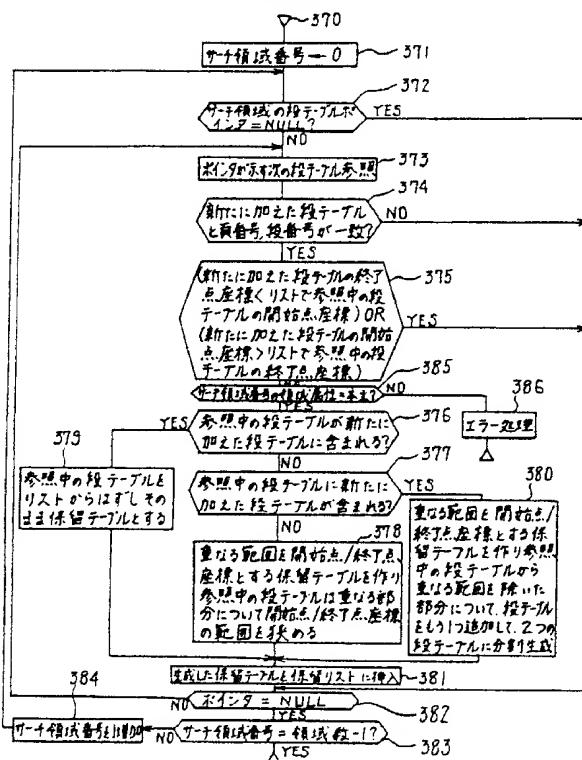


第11図

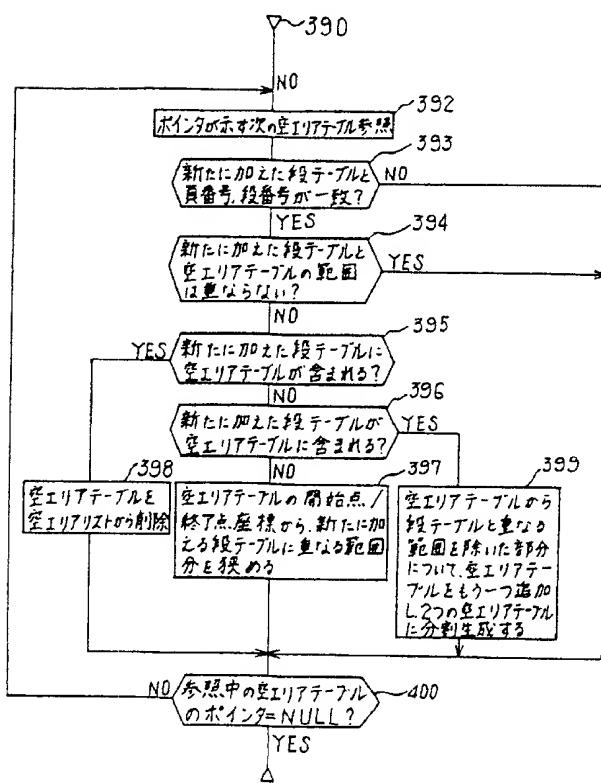


保留在リスト 140

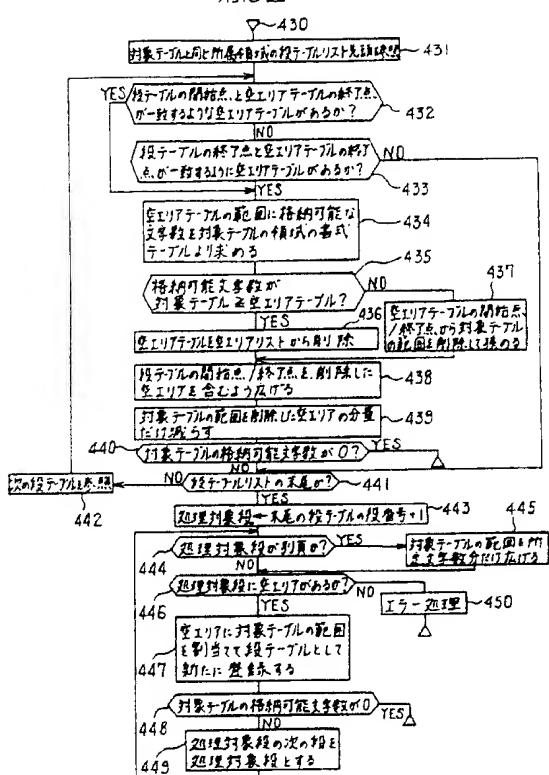
第13図



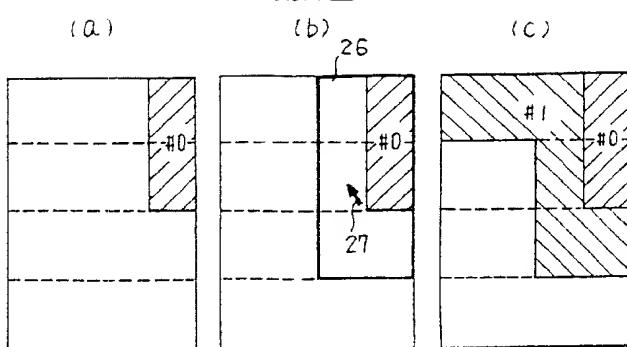
第14図



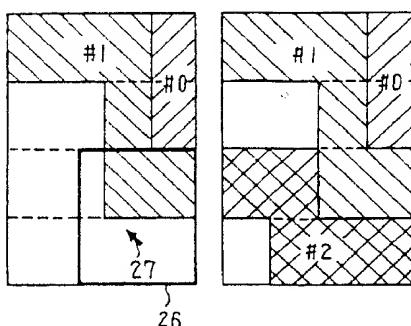
第15回



九〇六



(e)



卷之三

